R E P B L I Q U E F R A N Ç A S E

PCT/FR00/02220

INPI INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE

09/806756 04 OCT. 2000

FR00/02220 8-7-01

BREVET D'INVENTION

CERTIFICAT D'UTILITÉ - CERTIFICAT D'ADDITION

COPIE OFFICIELLE

Le Directeur général de l'Institut national de la propriété industrielle certifie que le document ci-annexé est la copie certifiée conforme d'une demande de titre de propriété industrielle déposée à l'Institut.

4

Fait à Paris, le 0 1 AOUT 2000

Pour le Directeur général de l'Institut
national de la propriété industrielle
Le Chef du Département des brevets

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN COMPLIANCE WITH RULE 17.1(a) OR (b)

Martine PLANCHE

INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE
SIEGE
26 bis, rue de Saint Petersbourg
75800 PARIS Cédex 08
Téléphone : 01 53 04 53 04
Télécopie : 01 42 93 59 30

ETABLISSEMENT PUBLIC NATIONAL CREE PAR LA LOI N° 51-444 DU 19 AVRIL 1951

INPI
INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE

26 bis, rue de Saint Pétersbourg
75800 Paris Cedex 08
Telephone : 01 53 04 53 04 Télécopie : 01 42 93 59 30

Code de la propriété intellectuelle-Livre VI

REQUÊTE EN DÉLIVRANCE

Confirmation d'un dépôt par télécopie

Cet imprimé est à remplir à l'encre noire en lettres capitales

N° 55 -1328

Réservé à l'INPI

DATE DE REMISE DES PIECES : 4 AOUT 199
N° D'ENREGISTREMENT NATIONAL : 991012
DEPARTEMENT DE DÉPÔT : 75 INPI PARIS
DATE DE DÉPÔT : 0 4 AOUT 1999

1 NOM ET ADRESSE DU DEMANDEUR OU DU MANDATAIRE À QUI LA CORRESPONDANCE DOIT ÊTRE ADRESSÉE

Cabinet PLASSERAUD
84, rue d'Amsterdam
75440 PARIS CEDEX 09

n°du pouvoir permanent : références du correspondant : BLo/EV/BFF990195 téléphone : 01 44 63 41 11

2 DEMANDE Nature du titre de propriété industrielle

[X] brevet d'invention [ ] demande divisionnaire [ ] demande initiale
[ ] certificat d'utilité [ ] transformation d'une demande de brevet europeen [ ] brevet d'invention [ ] certificat d'utilité n° date

Établissement du rapport de recherche [ ] différé [X] immédiat

Le demandeur, personne physique, requiert le paiement échelonné de la redevance [ ] oui [ ] non

Titre de l'invention (200 caractères maximum)

Procédé et dispositif de détection d'activité vocale

3 DEMANDEUR (S) n° SIREN 5 5 2 1 5 0 7 2 4 code APE-NAF

Nom et prénoms (souligner le nom patronymique) ou dénomination

MATRA NORTEL COMMUNICATIONS

Forme juridique

Société par Actions Simplifiée

Nationalité (s) Française

Adresse (s) complète (s) Pays

50, rue du Président Sadate
29100 QUIMPER

FRANCE

En cas d'insuffisance de place, poursuivre sur papier libre [ ]

4 INVENTEUR (S) Les inventeurs sont les demandeurs [ ] oui [X] non Si la réponse est non, fournir une désignation séparée

5 RÉDUCTION DU TAUX DES REDEVANCES [ ] requise pour la 1ère fois [ ] requise antérieurement au dépôt : joindre copie de la décision d'admission

6 DÉCLARATION DE PRIORITÉ OU REQUÊTE DU BÉNÉFICE DE LA DATE DE DÉPÔT D'UNE DEMANDE ANTÉRIEURE

pays d'origine	numéro	date de dépôt	nature de la demande

7 DIVISIONS antérieures à la présente demande n° date n° date

8 SIGNATURE DU DEMANDEUR OU DU MANDATAIRE (nom et qualité du signataire)

B. LOISEL
94-0311
CABINET PLASSERAUD

SIGNATURE DU PRÉPOSÉ À LA RÉCEPTION SIGNATURE APRÈS ENREGISTREMENT DE LA DEMANDE À L'INPI

La loi n°78-17 du 6 janvier 1978 relative à l'informatique aux fichiers et aux libertés s'applique aux réponses faites à ce formulaire. Elle garantit un droit d'acces et de rectification pour les données vous concernant auprès de l'INPI.

INPI
INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE

BLo/EV/BFF990195

BREVET D'INVENTION, CERTIFICAT D'UTILITE

DÉSIGNATION DE L'INVENTEUR
(si le demandeur n'est pas l'inventeur ou l'unique inventeur)

DEPARTEMENT DES BREVETS

26bis, rue de Saint-Pétersbourg
75800 Paris Cédex 08
Tél. : 01 53 04 53 04 - Télécopie : 01 42 93 59 30

N° D'ENREGISTREMENT NATIONAL

9910128

TITRE DE L'INVENTION :
Procédé et dispositif de détection d'activité vocale

La Demanderesse : MATRA NORTEL COMMUNICATIONS

Ayant pour Mandataire :

LE(S) SOUSSIGNÉ(S)

Cabinet PLASSERAUD
84, rue d'Amsterdam
75440 PARIS CEDEX 09

DÉSIGNE(NT) EN TANT QU'INVENTEUR(S) (indiquer nom, prénoms, adresse et souligner le nom patronymique) :

1/ **LUBIARZ** Stéphane
4, avenue Léon Heuzey
75016 PARIS
France

2/ **HINARD** Edouard
26, rue de la Fédération
75015 PARIS
France

3/ **CAPMAN** François
47, rue des Etats Généraux
78000 VERSAILLES
France

4/ **LOCKWOOD** Philip
10, rue de l'Amazone
95490 VAUREAL
France

NOTA : A titre exceptionnel, le nom de l'inventeur peut être suivi de celui de la société à laquelle il appartient (société d'appartenance) lorsque celle-ci est différente de la société déposante ou titulaire.

Date et signature (s) du (des) demandeur (s) ou du mandataire

Paris, le 4 août 1999

B. LOISEL
94-0311

CABINET PLASSERAUD

BA 113/140897

BEST AVAILABLE COPY

DOCUMENT COMPORTANT DES MODIFICATIONS

PAGE(S) DE LA DESCRIPTION OU DES REVENDICATIONS OU PLANCHE(S) DE DESSIN			R.M.*	DATE DE LA CORRESPONDANCE	TAMPON DATEUR DU CORRECTEUR
Modifiée(s)	Supprimée(s)	Ajoutée(s)			
7				25/01/00	27/01/00 - AL
19			RM	25/01/00	27/01/00 - AL

Un changement apporté à la rédaction des revendications d'origine, sauf si celui-ci découle des dispositions de l'article R.612-36 du code de la Propriété Intellectuelle, est signalé par la mention «R.M.» (revendications modifées).

PROCEDE ET DISPOSITIF DE DETECTION D'ACTIVITE VOCALE

La présente invention concerne les techniques numériques de traitement de signaux de parole. Elle concerne plus particulièrement les techniques faisant appel à une détection d'activité vocale afin d'effectuer des traitements différenciés selon que le signal supporte ou non une activité vocale.

Les techniques numériques en question relèvent de domaines variés : codage de la parole pour la transmission ou le stockage, reconnaissance de la parole, diminution du bruit, annulation d'écho...

Les méthodes de détection d'activité vocale ont pour principale difficulté la distinction entre l'activité vocale et le bruit qui accompagne le signal de parole.

Le document WO99/14737 décrit un procédé de détection d'activité vocale dans un signal de parole numérique traité par trames successives, dans lequel on procède à un débruitage a priori du signal de parole de chaque trame sur la base d'estimations du bruit obtenues lors du traitement d'une ou plusieurs trames précédentes, et on analyse les variations d'énergie du signal débruité a priori pour détecter un degré d'activité vocale de la trame. Le fait de procéder à la détection d'activité vocale sur la base d'un signal débruité a priori améliore sensiblement les performances de cette détection lorsque le bruit environnant est relativement important.

Dans les méthodes habituellement utilisées pour détecter l'activité vocale, les variations d'énergie du signal (direct ou débruité) sont analysées par rapport à une moyenne à long terme de l'énergie de ce signal, une augmentation relative de l'énergie instantanée suggérant l'apparition d'une activité vocale.

Un but de la présente invention est de proposer un autre type d'analyse permettant une détection d'activité vocale robuste au bruit pouvant accompagner le signal de parole.

Selon l'invention, il est proposé un procédé de détection d'activité vocale dans un signal de parole numérique dans au moins une bande de fréquences, suivant lequel on détecte l'activité vocale sur la base d'une analyse comprenant une comparaison, dans ladite bande de fréquences, de deux versions différentes du signal de parole dont l'une au moins est une version débruitée.

Ce procédé peut être exécuté sur toute la bande de fréquence du

signal, ou par sous-bandes, en fonction des besoins de l'application utilisant la détection d'activité vocale.

L'activité vocale peut être détectée de manière binaire pour chaque bande, ou mesurée par un paramètre variant continûment et pouvant résulter de la comparaison entre les deux versions différentes du signal de parole.

La comparaison porte typiquement sur des énergies respectives, évaluées dans ladite bande de fréquences, des deux versions différentes du signal de parole, ou sur une fonction monotone de ces énergies.

Un autre aspect de la présente invention se rapporte à un dispositif de détection d'activité vocale dans un signal de parole, comprenant des moyens de traitement de signal agencés pour mettre en œuvre un procédé tel que défini ci-dessus.

L'invention se rapporte encore à un programme d'ordinateur, chargeable dans une mémoire associée à un processeur, et comprenant des portions de code pour la mise en œuvre d'un procédé tel que défini ci-dessus lors de l'exécution dudit programme par le processeur, ainsi qu'à un support informatique, sur lequel est enregistré un tel programme.

D'autres particularités et avantages de la présente invention apparaîtront dans la description ci-après d'exemples de réalisation non limitatifs, en référence aux dessins annexés, dans lesquels :

- la figure 1 est un schéma synoptique d'une chaîne de traitement de signal utilisant un détecteur d'activité vocale selon l'invention ;
- la figure 2 est un schéma synoptique d'un exemple de détecteur d'activité vocale selon l'invention ;
- les figures 3 et 4 sont des organigrammes d'opérations de traitement de signal effectuées dans le détecteur de la figure 2,
- la figure 5 est un graphique montrant un exemple d'évolution d'énergies calculées dans le détecteur de la figure 2 et illustrant le principe de la détection d'activité vocale ;
- la figure 6 est un diagramme d'un automate de détection mis en œuvre dans le détecteur de la figure 2 ;
- la figure 7 est un schéma synoptique d'une autre réalisation d'un détecteur d'activité vocale selon l'invention ;
- la figure 8 est un organigramme d'opérations de traitement de signal effectuées dans le détecteur de la figure 7 ;

- la figure 9 est un graphique d'une fonction utilisée dans les opérations de la figure 8.

Le dispositif de la figure 1 traite un signal numérique de parole s. La chaîne de traitement de signal représentée produit des décisions d'activité vocale $\delta_{n,j}$ utilisables de façon connue en soi par des unités d'application, non représentées, assurant des fonctions telles que codage de la parole, reconnaissance de la parole, diminution du bruit, annulation d'écho... Les décisions $\delta_{n,j}$ peuvent comporter une résolution en fréquence (index j), ce qui permet d'enrichir des applications fonctionnant dans le domaine fréquentiel.

Un module de fenêtrage 10 met le signal s sous forme de fenêtres ou trames successives d'index n, constituées chacune d'un nombre N d'échantillons de signal numérique. De façon classique, ces trames peuvent présenter des recouvrements mutuels. Dans la suite de la présente description, on considérera, sans que ceci soit limitatif, que les trames sont constituées de N = 256 échantillons à une fréquence d'échantillonnage F_e de 8 kHz, avec une pondération de Hamming dans chaque fenêtre, et des recouvrements de 50 % entre fenêtres consécutives.

La trame de signal est transformée dans le domaine fréquentiel par un module 11 appliquant un algorithme classique de transformée de Fourier rapide (TFR) pour calculer le module du spectre du signal. Le module 11 délivre alors un ensemble de N = 256 composantes fréquentielles du signal de parole, notées $S_{n,f}$, où n désigne le numéro de la trame courante, et f une fréquence du spectre discret. Du fait des propriétés des signaux numériques dans le domaine fréquentiel, seuls les N/2 = 128 premiers échantillons sont utilisés.

Pour calculer les estimations du bruit contenu dans le signal s, on n'utilise pas la résolution fréquentielle disponible en sortie de la transformée de Fourier rapide, mais une résolution plus faible, déterminée par un nombre I de sous-bandes de fréquences couvrant la bande $[0, F_e/2]$ du signal. Chaque sous-bande i ($1 \leq i \leq I$) s'étend entre une fréquence inférieure f(i–1) et une fréquence supérieure f(i), avec f(0) = 0, et f(I) = F /2. Ce découpage en sous-bandes peut être uniforme ($f(i)–f(i–1) = F_e/2I$). Il peut ´galement être non uniforme (par exemple selon une échelle de barks). Un module 12 calcule les moyennes respectives des composantes spectrales $S_{n,f}$ du signal de parole

par sous-bandes, par exemple par une pondération uniforme telle que :

$$S_{n,i} = \frac{1}{f(i) - f(i-1)} \sum_{f \in [f(i-1), f(i)[} S_{n,f}$$

Ce moyennage diminue les fluctuations entre les sous-bandes en moyennant les contributions du bruit dans ces sous-bandes, ce qui diminuera la variance de l'estimateur de bruit. En outre, ce moyennage permet de diminuer la complexité du système.

Les composantes spectrales moyennées $S_{n,i}$ sont adressées à un module 15 de détection d'activité vocale et à un module 16 d'estimation du bruit. On note $\hat{B}_{n,i}$ l'estimation à long terme de la composante de bruit produite par le module 16 relativement à la trame n et à la sous-bande i.

Ces estimations à long terme $\hat{B}_{n,i}$ peuvent par exemple être obtenues de la manière décrite dans WO99/14737. On peut aussi utiliser un simple lissage au moyen d'une fenêtre exponentielle définie par un facteur d'oubli λ_B :

$$\hat{B}_{n,i} = \lambda_B . \hat{B}_{n-1,i} + (1-\lambda_B) . S_{n,i}$$

avec λ_B égal à 1 si le détecteur d'activité vocale 15 indique que la sous-bande i porte une activité vocale, et égal à une valeur comprise entre 0 et 1 sinon.

Bien entendu, il est possible d'utiliser d'autres estimations à long terme représentatives de la composante de bruit comprise dans le signal de parole, ces estimations peuvent représenter une moyenne à long terme, ou encore un minimum de la composante $S_{n,i}$ sur une fenêtre glissante suffisamment longue.

Les figures 2 à 6 illustrent une première réalisation du détecteur d'activité vocale 15. Un module de débruitage 18 exécute, pour chaque trame n et chaque sous-bande i, les opérations correspondant aux étapes 180 à 187 de la figure 3, pour produire deux versions débruitées $\hat{E}p_{1,n,i}$, $\hat{E}p_{2,n,i}$ du signal de parole. Ce débruitage est opéré par soustraction spectrale non-linéaire. La première version $\hat{E}p_{1,n,i}$ est débruitée de façon à ne pas être inférieure, dans le domaine spectral, à une fraction $\beta 1_i$ de l'estimation à long terme $\hat{B}_{n-\tau 1,i}$. La seconde version $\hat{E}p_{2,n,i}$ est débruitée de façon à ne pas être inférieure, dans le domaine spectral, à une fraction $\beta 2_i$ de l'estimation à long terme $\hat{B}_{n-\tau 1,i}$. La quantité $\tau 1$ est un retard exprimé en nombre de trames, qui peut être fixe (par

exemple $\tau 1 = 1$) ou variable. Il est d'autant faible qu'on est confiant dans la détection d'activité vocale. Les fractions $\beta 1_i$ et $\beta 2_i$ (telles que $\beta 1_i > \beta 2_i$) peuvent être dépendantes ou indépendantes de la sous-bande i. Des valeurs préférées correspondent pour $\beta 1_i$ à une atténuation de 10 dB, et pour $\beta 2_i$ à une atténuation de 60 dB, soit $\beta 1_i \approx 0{,}3$ et $\beta 2_i \approx 0{,}001$.

A l'étape 180, le module 18 calcule, avec la résolution des sous-bandes i, la réponse en fréquence $Hp_{n,i}$ du filtre de débruitage a priori, selon :

$$Hp_{n,i} = \frac{S_{n,i} - \alpha'_{n-\tau 1,i} \cdot \hat{B}_{n-\tau 1,i}}{S_{n-\tau 2,i}}$$

où $\tau 2$ est un retard entier positif ou nul et $\alpha'_{n,i}$ est un coefficient de surestimation du bruit. Ce coefficient de surestimation $\alpha'_{n,i}$ peut être dépendant ou indépendant de l'index de trame n et/ou de l'index de sous-bande i. Dans une réalisation préférée, il dépend à la fois de n et i, et il est déterminé comme décrit dans le document WO99/14737. Un premier débruitage est effectué à l'étape 181 : $\hat{E}p_{n,i} = Hp_{n,i} \cdot S_{n,i}$. Aux étapes 182 à 184, les composantes spectrales $\hat{E}p_{1,n,i}$ sont calculées selon $\hat{E}p_{1,n,i} = \max\left(\hat{E}p_{n,i} ; \beta 1_i \cdot \hat{B}_{n-\tau 1,i}\right)$, et aux étapes 182 à 184, les composantes spectrales $\hat{E}p_{2,n,i}$ sont calculées selon $\hat{E}p_{2,n,i} = \max\left(\hat{E}p_{n,i} ; \beta 2_i \cdot \hat{B}_{n-\tau 1,i}\right)$.

Le détecteur d'activité vocale 15 de la figure 2 comporte un module 19 qui calcule des énergies des versions débruitées du signal $\hat{E}p_{1,n,i}$ et $\hat{E}p_{2,n,i}$, respectivement comprises dans m bandes de fréquences désignées par l'index j ($1 \leq j \leq m$, $m \geq 1$). Cette résolution peut être la même que celle des sous-bandes définies par le module 12 (index i), ou une résolution moins fine pouvant aller jusqu'à l'ensemble de la bande utile $[0, F_e/2]$ du signal (cas $m = 1$). A titre d'exemple, le module 12 peut définir $I = 16$ sous-bandes uniformes de la bande $[0, F_e/2]$, et le module 19 peut conserver $m = 3$ bandes plus larges, chaque bande d'index j couvrant les sous-bandes d'index i allant de imin(j) à imax(j), avec imin(1) = 1, imin(j+1) = imax(j) + 1 pour $1 \leq j < m$, et imax(m) = I. A l'étape 190 (figure 3), le module 19 calcule les énergies par bande :

$$E_{1,n,j} = \sum_{i=imin(j)}^{imax(j)} [f(i) - f(i-1)].\hat{E}p_{1,n,i}^2$$

$$E_{2,n,j} = \sum_{i=imin(j)}^{imax(j)} [f(i) - f(i-1)].\hat{E}p_{2,n,i}^2$$

Un module 20 du détecteur d'activité vocale 15 effectue un lissage temporel des énergies $E_{1,n,j}$ et $E_{2,n,j}$ pour chacune des bandes d'index j, ce qui correspond aux étapes 200 à 205 de la figure 4. Le lissage de ces deux énergies est effectué au moyen d'une fenêtre de lissage déterminée en comparant l'énergie $E_{2,n,j}$ de la version la plus débruitée à son énergie lissée précédemment calculée $\overline{E}_{2,n-1,j}$, ou à une valeur de l'ordre de cette énergie lissée $\overline{E}_{2,n-1,j}$ (tests 200 et 201). Cette fenêtre de lissage peut être une fenêtre exponentielle définie par un facteur d'oubli λ compris entre 0 et 1. Ce facteur d'oubli λ peut prendre trois valeurs : l'une λ_r très proche de 0 (par exemple $\lambda_r = 0$) choisie à l'étape 202 si $E_{2,n,j} \leq \overline{E}_{2,n-1,j}$; la seconde λ_q très proche de 1 (par exemple $\lambda_q = 0{,}99999$) choisie à l'étape 203 si $E_{2,n,j} > \Delta.\overline{E}_{2,n-1,j}$, Δ étant un coefficient plus grand que 1 ; et la troisième λ_p comprise entre 0 et λ_q (par exemple $\lambda_p = 0{,}98$) choisie à l'étape 204 si $\overline{E}_{2,n-1,j} < E_{2,n,j} \leq \Delta.\overline{E}_{2,n-1,j}$. Le lissage exponentiel avec le facteur d'oubli λ est ensuite effectué classiquement à l'étape 205 selon :

$$\overline{E}_{1,n,j} = \lambda.\overline{E}_{1,n-1,j} + (1-\lambda).E_{1,n,j}$$

$$\overline{E}_{2,n,j} = \lambda.\overline{E}_{2,n-1,j} + (1-\lambda).E_{2,n,j}$$

Un exemple de variation dans le temps des énergies $E_{1,n,j}$, $E_{2,n,j}$ et des énergies lissées $\overline{E}_{1,n,j}$ et $\overline{E}_{2,n,j}$ est représenté sur la figure 5. On voit qu'on arrive à un bon suivi des énergies lissées lorsqu'on détermine le facteur d'oubli sur la base des variations de l'énergie $E_{2,n,j}$ correspondant à la version la plus débruitée du signal. Le facteur d'oubli λ_p permet de prendre en compte les augmentations de niveau du bruit de fond, les diminutions d'énergie étant suivies par le facteur d'oubli λ_r. Le facteur d'oubli λ_q très proche de 1 fait que les énergies lissées ne suivent pas les augmentations d'énergies brusques

dues à la parole. Le facteur λ_q reste toutefois légèrement inférieur à 1 pour éviter les erreurs causées par une augmentation du bruit de fond pouvant survenir pendant une assez longue période de parole.

L'automate de détection d'activité vocale est contrôlé notamment par un paramètre résultant d'une comparaison des énergies $E_{1,n,j}$ et $E_{2,n,j}$. Ce paramètre peut notamment être le rapport $d_{n,j} = E_{1,n,j}/E_{2,n,j}$. On voit sur la figure 5 que ce rapport $d_{n,j}$ permet de bien détecter les phases de parole (représentées par des hachures).

Le contrôle de l'automate de détection peut également utiliser d'autres paramètres, tels qu'un paramètre lié au rapport signal-sur-bruit : $snr_{n,j} = E_{1,n,j}/\overline{E}_{1,n,j}$. Le module 21 de contrôle des automates relatifs aux différentes bandes d'index j calcule les paramètres $d_{n,j}$ et $snr_{n,j}$ à l'étape 210, puis détermine l'état des automates. Le nouvel état $\delta_{n,j}$ de l'automate relatif à la bande j dépend de l'état précédent $\delta_{n-1,j}$, de $d_{n,j}$ et de $snr_{n,j}$, par exemple comme indiqué sur le diagramme de la figure 6.

Quatre états sont possibles : $\delta_j = 0$ détecte le silence, ou absence de parole ; $\delta_j=2$ détecte la présence d'une activité vocale ; et les états $\delta_j = 1$ et $\delta_j = 3$ sont des états intermédiaires de montée et de descente. Lorsque l'automate est dans l'état de silence ($\delta_{n-1,j} = 0$), il y reste si $d_{n,j}$ dépasse un premier seuil $\alpha 1_j$, et il passe dans l'état de montée dans le cas contraire. Dans l'état de montée ($\delta_{n-1,j} = 1$), il revient dans l'état de silence si $d_{n,j}$ dépasse un second seuil $\alpha 2_j$; et il passe dans l'état de parole dans le cas contraire. Lorsque l'automate est dans l'état de parole ($\delta_{n-1,j} = 2$), il y reste si $snr_{n,j}$ dépasse un troisième seuil $\alpha 3_j$, et il passe dans l'état de descente dans le cas contraire. Dans l'état de descente ($\delta_{n-1,j} = 3$), l'automate revient dans l'état de parole si $snr_{n,j}$ dépasse un quatrième seuil $\alpha 4_j$, et il revient dans l'état de silence dans le cas contraire. Les seuils $\alpha 1_j$, $\alpha 2_j$, $\alpha 3_j$ et $\alpha 4_j$ peuvent être optimisés séparément pour chacune des bandes de fréquences j.

Il est également possible que le module 21 fasse interagir les automates relatifs aux différentes bandes.

En particulier, il peut forcer à l'état de parole les automates relatifs à

chacune des sous-bandes dès lors que l'un d'entre eux se trouve dans l'état de parole. Dans ce cas, la sortie du détecteur d'activité vocale 15 concerne l'ensemble de la bande du signal.

Les deux annexes à la présente description montrent un code source en langage C++, avec une représentation des données en virgule fixe, correspondant à une mise en œuvre de l'exemple de procédé de détection d'activité vocale décrit ci-dessus. Pour réaliser le détecteur, une possibilité est de traduire ce code source en code exécutable, de l'enregistrer dans une mémoire de programme associée à un processeur de traitement de signal approprié, et de le faire exécuter par ce processeur sur les signaux d'entrée du détecteur. La fonction *a_priori_signal_power* présentée en annexe 1 correspond aux opérations incombant aux modules 18 et 19 du détecteur d'activité vocale 15 de la figure 2. La fonction *voice_activity_detector* présentée en annexe 2 correspond aux opérations incombant aux modules 20 et 21 de ce détecteur.

Dans l'exemple particulier des annexes, les paramètres suivant ont été employés : $\tau 1 = 1$; $\tau 2 = 0$; $\beta 1_i = 0{,}3$; $\beta 2_i = 0{,}001$; $m = 3$; $\Delta = 4{,}953$; $\lambda_p = 0{,}98$; $\lambda_q = 0{,}99999$; $\lambda_r = 0$; $\alpha 1_j = \alpha 2_j = \alpha 4_j = 1{,}221$; $\alpha 3_j = 1{,}649$. Le Tableau I ci-après donne les correspondances entre les notations employées dans la précédente description et dans les dessins et celles employées dans l'annexe.

subband	i
E[subband]	$S_{n,i}$
module	$\hat{E}p_{n,i}$ ou $\hat{E}p_{1,n,i}$ ou $\hat{E}p_{2,n,i}$
param.beta_a_priori1	$\beta 1_i$
param.beta_a_priori2	$\beta 2_i$
vad	j–1
param.vad_number	m
P1[vad]	$E_{1,n,j-1}$
P1s[vad]	$\overline{E}_{1,n,j-1}$
P2[vad]	$E_{2,n,j-1}$
P2s[vad]	$\overline{E}_{2,n,j-1}$
DELTA_P	$Log(\Delta)$
d	$Log(d_{n,j})$
snr	$Log(snr_{n,j})$
NOISE	état de silence
ASCENT	état de montée
SIGNAL	état de parole
DESCENT	état de descente
D_NOISE	$Log(\alpha 1_j)$
D_SIGNAL	$Log(\alpha 2_j)$
SNR_SIGNAL	$Log(\alpha 3_j)$
SNR_NOISE	$Log(\alpha 4_j)$

TABLEAU I

Dans la variante de réalisation illustrée par la figure 7, le module de débruitage 25 du détecteur d'activité vocale 15 délivre une seule version débruitée $\hat{E}p_{n,i}$ du signal de parole, pour que le module 26 en calcule l'énergie $E_{2,n,j}$ pour chaque bande j. L'autre version dont le module 26 calcule l'énergie est directement représentée par les échantillons non débruités $S_{n,i}$.

Comme précédemment, diverses méthodes de débruitage peuvent être

appliquées par le module 25. Dans l'exemple illustré par les étapes 250 à 256 de la figure 8, le débruitage est opéré par soustraction spectrale non-linéaire avec un coefficient de surestimation du bruit dépendant d'une quantité ρ liée au rapport signal-sur-bruit. Aux étapes 250 à 252, un débruitage préliminaire est effectué pour chaque sous-bande d'index i selon :

$$S'_{n,i} = \max\left(S_{n,i} - \alpha.\hat{B}_{n-1,i}\,;\, \beta.\hat{B}_{n-1,i}\right),$$

le coefficient de surestimation préliminaire étant par exemple $\alpha = 2$, et la fraction β pouvant correspondre à une atténuation du bruit de l'ordre de 10 dB.

La quantité ρ est prise égale au rapport $S'_{n,i}/S_{n,i}$ à l'étape 253. Le facteur de surestimation $f(\rho)$ varie de façon non-linéaire avec la quantité ρ, par exemple comme représenté sur la figure 9. Pour les valeurs de ρ les plus proches de 0 ($\rho < \rho_1$), le rapport signal-sur-bruit est faible, et on peut prendre un facteur de surestimation $f(\rho) = 2$. Pour les valeurs les plus élevées de ρ ($\rho_2 \leq \rho \leq 1$), le bruit est faible et n'a pas besoin d'être surestimé ($f(\rho)=1$). Entre ρ_1 et ρ_2, $f(\rho)$ décroît de 2 à 1, par exemple linéairement. Le débruitage proprement dit, fournissant la version $\hat{E}p_{n,i}$, est effectué aux étapes 254 à 256 :

$$\hat{E}p_{n,i} = \max\left(S_{n,i} - f(\rho).\hat{B}_{n-1,i}\,;\, \beta.\hat{B}_{n-1,i}\right).$$

Le détecteur d'activité vocale 15 considéré en référence à la figure 7 utilise, dans chaque bande de fréquences d'index j (et/ou en pleine bande), un automate de détection à deux états, silence ou parole. Les énergies $E_{1,n,j}$ et $E_{2,n,j}$ calculées par le module 26 sont respectivement celles contenues dans les composantes $S_{n,i}$ du signal de parole et celles contenues dans les composantes débruitées $\hat{E}p_{n,i}$ calculées sur les différentes bandes comme indiqué à l'étape 260 de la figure 8. La comparaison des deux versions différentes du signal de parole porte sur des différences respectives entre les énergies $E_{1,n,j}$ et $E_{2,n,j}$ et un minorant de l'énergie $E_{2,n,j}$ de la version débruitée.

Ce minorant $E_{2min,j}$ peut notamment correspondre à une valeur minimale, sur une fenêtre glissante, de l'énergie $E_{2,n,j}$ de la version débruitée du signal de parole dans la bande de fréquences considérée. Dans ce cas, un

module 27 stocke dans une mémoire de type premier entré - premier sorti (FIFO) les L valeurs les plus récentes de l'énergie $E_{2,n,j}$ du signal débruité dans chaque bande j, sur une fenêtre glissante représentant par exemple de l'ordre de 20 trames, et délivre les énergies minimales $E_{2min,j} = \min_{0 \le k < L} E_{2,n-k,j}$ sur cette fenêtre (étape 270 de la figure 8). Dans chaque bande, cette énergie minimale $E_{2min,j}$ sert de minorant pour le module 28 de contrôle de l'automate de détection, qui utilise une mesure M_j donnée par $M_j = \frac{E_{2,n,j} - E_{2min,j}}{E_{1,n,j} - E_{2min,j}}$ (étape 280).

L'automate peut être un simple automate binaire utilisant un seuil A_j, dépendant éventuellement de la bande considérée : si $M_j > A_j$, le bit de sortie $\delta_{n,j}$ du détecteur représente un état de silence pour la bande j, et si $M_j \le A_j$, il représente un état de parole. En variante, le module 28 pourrait délivrer une mesure non binaire de l'activité vocale, représentée par une fonction décroissante de M_j.

En variante, le minorant $E_{2min,j}$ utilisé à l'étape 280 pourrait être calculé à l'aide d'une fenêtre exponentielle, avec un facteur d'oubli. Il pourrait aussi être représenté par l'énergie sur la bande j de la quantité $\beta.\hat{B}_{n-1,i}$ servant de plancher dans le débruitage par soustraction spectrale.

Dans ce qui précède, l'analyse effectuée pour décider de la présence ou de l'absence d'activité vocale porte directement sur des énergies de versions différentes du signal de parole. Bien entendu, les comparaisons pourraient porter sur une fonction monotone de ces énergies, par exemple un logarithme, ou sur une quantité ayant un comportement analogue aux énergies selon l'activité vocale (par exemple la puissance).

ANNEXE 1

```
/*************************************************************************
******
 * description
 * ~~~~~~~~~~~
 * NSS module:
 *  signal power before VAD
 *
 *************************************************************************
******/

/*-----------------------------------------------------------------------
-----*
 *                          included files
 *-----------------------------------------------------------------------
-----*/
#include <assert.h>

#include "private.h"


/*-----------------------------------------------------------------------
-----*
 *                            private
 *-----------------------------------------------------------------------
-----*/
Word32 power(Word16 module, Word16 beta, Word16 thd, Word16 val);


/*-----------------------------------------------------------------------
-----*
 *                            a_priori_signal_power
 *-----------------------------------------------------------------------
-----*/
void a_priori_signal_power
(
/* IN */     Word16 *E, Word16 *internal_state, Word16 *max_noise, W
ord16 *long_term_noise,
             Word16 *frequential_scale,

/* IN&OUT */ Word16 *alpha,

/* OUT */    Word32 *P1, Word32 *P2
)
{
   int vad;

   for(vad = 0; vad < param.vad_number; vad++) {
      int start = param.vads[vad].first_subband_for_power;
      int stop = param.vads[vad].last_subband;
      int subband;
      int uniform_subband;

      uniform_subband = 1;
```

```
      for(subband = start; subband <= stop; subband++)
         if(param.subband_size[subband] != param.subband_size[start]
)
            uniform_subband = 0;

      P1[vad] = 0; move32();
      P2[vad] = 0; move32();
      test(); if(sub(internal_state[vad], NOISE) == 0) {
         for(subband = start; subband <= stop; subband++) {
            Word32 pwr;
            Word16 shift;
            Word16 module;
            Word16 alpha_long_term;

            alpha_long_term = shr(max_noise[subband], 2); move16();
            test(); test(); if(sub(alpha_long_term, long_term_noise[
subband]) >= 0) {
               alpha[subband] = 0x7fff; move16();
               alpha_long_term = long_term_noise[subband]; move16();
            } else if(sub(max_noise[subband], long_term_noise[subban
d]) < 0) {
               alpha[subband] = 0x2000; move16();
               alpha_long_term = shr(long_term_noise[subband],2); mo
ve16();
            } else {
               alpha[subband] = div_s(alpha_long_term, long_term_noi
se[subband]); move16();
            }
            module = sub(E[subband], shl(alpha_long_term, 2)); move1
6();

            if(uniform_subband) {
               shift = shl(frequential_scale[subband], 1); move16();
            } else {
               shift = add(param.subband_shift[subband], shl(frequen
tial_scale[subband], 1)); move16();
            }

            pwr = power(module, param.beta_a_priori1, long_term_nois
e[subband], long_term_noise[subband]);
            pwr = L_shr(pwr, shift);
            P1[vad] = L_add(P1[vad], pwr); move32();

            pwr = power(module, param.beta_a_priori2, long_term_nois
e[subband], long_term_noise[subband]);
            pwr = L_shr(pwr, shift);
            P2[vad] = L_add(P2[vad], pwr); move32();
         }
      } else {
         for(subband = start; subband <= stop; subband++) {
            Word32 pwr;
            Word16 shift;
            Word16 module;
            Word16 alpha_long_term;

            alpha_long_term = mult(alpha[subband], long_term_noise[s
```

```
ubband]); move16();
            module = sub(E[subband], shl(alpha_long_term, 2)); move1
6();

            if(uniform_subband) {
               shift = shl(frequential_scale[subband], 1); move16();
            } else {
               shift = add(param.subband_shift[subband], shl(frequen
tial_scale[subband], 1)); move16();
            }

            pwr = power(module, param.beta_a_priori1, long_term_nois
e[subband], E[subband]);
            pwr = L_shr(pwr, shift);
            P1[vad] = L_add(P1[vad], pwr); move32();

            pwr = power(module, param.beta_a_priori2, long_term_nois
e[subband], E[subband]);
            pwr = L_shr(pwr, shift);
            P2[vad] = L_add(P2[vad], pwr); move32();
         }
      }
   }
}


/*-------------------------------------------------------------------
-----*
 *                                  power
 *-------------------------------------------------------------------
-----*/
Word32 power(Word16 module, Word16 beta, Word16 thd, Word16 val)
{
   Word32 power;

   test(); if(sub(module, mult(beta, thd)) <= 0) {
      Word16 hi, lo;

      power = L_mult(val, val); move32();

      L_Extract(power, &hi, &lo);
      power = Mpy_32_16(hi, lo, beta); move32();

      L_Extract(power, &hi, &lo);
      power = Mpy_32_16(hi, lo, beta); move32();
   } else {
      power = L_mult(module, module); move32();
   }
   return(power);
}
```

ANNEXE 2

```
/*****************************************************************************
******
 * description
 * ~~~~~~~~~~~
 * NSS module:
 *  VAD
 *
 *****************************************************************************
******/

/*-----------------------------------------------------------------------
-----*
 *                              included files
 *-----------------------------------------------------------------------
-----*/
#include <assert.h>

#include "private.h"

#include "simutool.h"


/*-----------------------------------------------------------------------
-----*
 *                                 private
 *-----------------------------------------------------------------------
-----*/
#define DELTA_P                 (1.6 * 1024)
#define D_NOISE                 (.2 * 1024)
#define D_SIGNAL                (.2 * 1024)
#define SNR_SIGNAL              (.5 * 1024)
#define SNR_NOISE               (.2 * 1024)


/*-----------------------------------------------------------------------
-----*
 *                          voice_activity_detector
 *-----------------------------------------------------------------------
-----*/
void voice_activity_detector
(
/* IN */     Word32 *P1, Word32 *P2, Word16 frame_counter,

/* IN&OUT */ Word32 *P1s, Word32 *P2s, Word16 *internal_state,

/* OUT */    Word16 *state
)
{
   int vad;
   int signal;
   int noise;
```

```
   signal = 0; move16();
   noise = 1; move16();
   for(vad = 0; vad < param.vad_number; vad++) {
      Word16 snr, d;
      Word16 logP1, logP1s;
      Word16 logP2, logP2s;


      logP2  = logfix(P2[vad]); move16();
      logP2s = logfix(P2s[vad]); move16();

      test(); if(L_sub(P2[vad], P2s[vad]) > 0) {
         Word16 hi1, lo1;
         Word16 hi2, lo2;

         L_Extract(L_sub(P1[vad], P1s[vad]), &hi1, &lo1);
         L_Extract(L_sub(P2[vad], P2s[vad]), &hi2, &lo2);

         test(); if(sub(sub(logP2, logP2s), DELTA_P) < 0) {
            P1s[vad] = L_add(P1s[vad], L_shr(Mpy_32_16(hi1, lo1, 0x6
666), 4)); move32();
            P2s[vad] = L_add(P2s[vad], L_shr(Mpy_32_16(hi2, lo2, 0x6
666), 4)); move32();
         } else {
            P1s[vad] = L_add(P1s[vad], L_shr(Mpy_32_16(hi1, lo1, 0x6
8db), 13)); move32();
            P2s[vad] = L_add(P2s[vad], L_shr(Mpy_32_16(hi2, lo2, 0x6
8db), 13)); move32();
         }
      } else {
         P1s[vad] = P1[vad]; move32();
         P2s[vad] = P2[vad]; move32();
      }


      logP1  = logfix(P1[vad]); move16();
      logP1s = logfix(P1s[vad]); move16();


      d = sub(logP1, logP2); move16();
      snr = sub(logP1, logP1s); move16();

      ProbeFix16("d", &d, 1, 1.);
      ProbeFix16("_snr", &snr, 1, 1.);
{
   Word16 pp;
   ProbeFix16("p1", &logP1, 1, 1.);
   ProbeFix16("p2", &logP2, 1, 1.);
   ProbeFix16("p1s", &logP1s, 1, 1.);
   ProbeFix16("p2s", &logP2s, 1, 1.);
   pp = logP2 - logP2s;
   ProbeFix16("dp", &pp, 1, 1.);
}
```

```
   test(); if(sub(internal_state[vad], NOISE) == 0)
      goto LABEL_NOISE;
   test(); if(sub(internal_state[vad], ASCENT) == 0)
      goto LABEL_ASCENT;
   test(); if(sub(internal_state[vad], SIGNAL) == 0)
      goto LABEL_SIGNAL;
   test(); if(sub(internal_state[vad], DESCENT) == 0)
      goto LABEL_DESCENT;

   LABEL_NOISE:
   test(); if(sub(d, D_NOISE) < 0) {
      internal_state[vad] = ASCENT; move16();
   }
   goto LABEL_END_VAD;

   LABEL_ASCENT:
   test(); if(sub(d, D_SIGNAL) < 0) {
      internal_state[vad] = SIGNAL; move16();
      signal = 1; move16();
      noise = 0; move16();
   } else {
      internal_state[vad] = NOISE; move16();
   }
   goto LABEL_END_VAD;

   LABEL_SIGNAL:
   test(); if(sub(snr, SNR_SIGNAL) < 0) {
      internal_state[vad] = DESCENT; move16();
   } else {
      signal = 1; move16();
   }
   noise = 0; move16();
   goto LABEL_END_VAD;

   LABEL_DESCENT:
   test(); if(sub(snr, SNR_NOISE) < 0) {
      internal_state[vad] = NOISE; move16();
   } else {
      internal_state[vad] = SIGNAL; move16();
      signal = 1; move16();
      noise = 0; move16();
   }
   goto LABEL_END_VAD;

   LABEL_END_VAD:
   ;
}

*state = TRANSITION; move16();
test(); test(); if(signal != 0) {
   test(); if(sub(frame_counter, param.init_frame_number) >= 0) {
      for(vad = 0; vad < param.vad_number; vad++) {
         internal_state[vad] = SIGNAL; move16();
      }
      *state = SIGNAL; move16();
   }
```

```
    } else if(noise != 0) {
        *state = NOISE; move16();
    }
}
```

REVENDICATIONS

1. Procédé de détection d'activité vocale dans un signal de parole numérique (s) dans au moins une bande de fréquences, caractérisé en ce qu'on détecte l'activité vocale sur la base d'une analyse comprenant une comparaison, dans ladite bande de fréquences, de deux versions différentes du signal de parole dont l'une au moins est une version débruitée.

2. Procédé selon la revendication 1, dans lequel ladite comparaison porte sur des énergies respectives ($E_{1,n,j}$, $E_{2,n,j}$), évaluées dans ladite bande de fréquences, des deux versions différentes du signal de parole, ou sur une fonction monotone desdites énergies.

3. Procédé selon la revendication 1 ou 2, dans lequel ladite analyse comprend en outre un lissage temporel de l'énergie ($E_{1,n,j}$) d'une desdites versions du signal de parole, et une comparaison entre l'énergie de ladite version et l'énergie lissée ($\overline{E}_{1,n,j}$).

4. Procédé selon la revendication 3, dans lequel la comparaison entre l'énergie de ladite version ($E_{1,n,j}$) et l'énergie lissée ($\overline{E}_{1,n,j}$) contrôle les transitions d'un automate de détection d'activité vocale d'un état de parole vers un état de silence, tandis que la comparaison des deux versions différentes du signal de parole contrôle les transitions de l'automate de détection de l'état de silence vers l'état de parole.

5. Procédé selon l'une quelconque des revendications 1 à 4, dans lequel les deux versions différentes du signal de parole sont deux versions débruitées par soustraction spectrale non-linéaire, une première des deux versions ($\hat{E}p_{1,n,i}$) étant débruitée de façon à ne pas être inférieure, dans le domaine spectral, à une première fraction ($\beta 1_i$) d'une estimation à long terme ($\hat{B}_{n,i}$) représentative d'une composante de bruit comprise dans le signal de parole, et la seconde des deux versions ($\hat{E}p_{2,n,i}$) étant débruitée de façon à ne pas être inférieure, dans le domaine spectral, à une seconde fraction ($\beta 2_i$) de ladite estimation à long terme, plus grande que la première fraction.

6. Procédé selon la revendication 5, dans lequel on effectue un lissage temporel de l'énergie de chacune des deux versions du signal de parole, au moyen d'une fenêtre de lissage déterminée en comparant l'énergie ($E_{2,n,j}$) de la seconde des deux versions à l'énergie lissée ($\overline{E}_{2,n,j}$) de la seconde des deux versions.

7. Procédé selon la revendication 6, dans lequel la fenêtre de lissage est une fenêtre exponentielle définie par un facteur d'oubli (λ).

8. Procédé selon la revendication 7, dans lequel le facteur d'oubli (λ) a une valeur (λ_r) sensiblement nulle lorsque l'énergie ($E_{2,n,j}$) de la seconde des deux versions est inférieure à une valeur de l'ordre de l'énergie lissée ($\overline{E}_{2,n,j}$) de la seconde des deux versions.

9. Procédé selon la revendication 8, dans lequel le facteur d'oubli (λ) a une première valeur (λ_q) sensiblement égale à 1 lorsque l'énergie ($E_{2,n,j}$) de la seconde des deux versions est supérieure à ladite valeur de l'ordre de l'énergie lissée multipliée par un coefficient (Δ) plus grand que 1, et une seconde valeur (λ_p) comprise entre 0 et ladite première valeur lorsque l'énergie de la seconde des deux versions est supérieure à ladite valeur de l'ordre de l'énergie lissée et inférieure à ladite valeur de l'ordre de l'énergie lissée multipliée par ledit coefficient.

10. Procédé selon l'une quelconque des revendications 5 à 9, dans lequel les première et seconde fractions ($\beta 1_i$, $\beta 2_i$) correspondent sensiblement à des atténuations de 10 dB et de 60 dB, respectivement.

11. Procédé selon l'une quelconque des revendications 1 à 10, dans lequel la comparaison des deux versions différentes du signal de parole porte sur des différences respectives entre les énergies ($E_{1,n,j}$, $E_{2,n,j}$) de ces deux versions dans ladite bande de fréquences et un minorant ($E_{2min,j}$) de l'énergie ($E_{2,n,j}$) de la version débruitée du signal de parole dans ladite bande de fréquences.

12. Procédé selon la revendication 11, dans lequel l'une des deux versions différentes du signal de parole est une version non débruitée du signal de parole.

13. Dispositif de détection d'activité vocale dans un signal de parole, comprenant des moyens de traitement de signal (15) agencés pour mettre en œuvre un procédé selon l'une quelconque des revendications 1 à 12.

14. Programme d'ordinateur, chargeable dans une mémoire associée à un processeur, et comprenant des portions de code pour la mise en œuvre d'un procédé selon l'une quelconque des revendications 1 à 12 lors de l'exécution dudit programme par le processeur.

15. Support informatique, sur lequel est enregistré un programme selon la revendication 14.

dues à la parole. Le facteur λ_q reste toutefois légèrement inférieur à 1 pour éviter les erreurs causées par une augmentation du bruit de fond pouvant survenir pendant une assez longue période de parole.

L'automate de détection d'activité vocale est contrôlé notamment par un paramètre résultant d'une comparaison des énergies $E_{1,n,j}$ et $E_{2,n,j}$. Ce paramètre peut notamment être le rapport $d_{n,j} = E_{1,n,j}/E_{2,n,j}$. On voit sur la figure 5 que ce rapport $d_{n,j}$ permet de bien détecter les phases de parole (représentées par des hachures).

Le contrôle de l'automate de détection peut également utiliser d'autres paramètres, tels qu'un paramètre lié au rapport signal-sur-bruit : $snr_{n,j} = E_{1,n,j}/\overline{E}_{1,n,j}$, ce qui revient à prendre en compte une comparaison entre les énergies $E_{1,n,j}$ et $\overline{E}_{1,n,j}$. Le module 21 de contrôle des automates relatifs aux différentes bandes d'index j calcule les paramètres $d_{n,j}$ et $snr_{n,j}$ à l'étape 210, puis détermine l'état des automates. Le nouvel état $\delta_{n,j}$ de l'automate relatif à la bande j dépend de l'état précédent $\delta_{n-1,j}$, de $d_{n,j}$ et de $snr_{n,j}$, par exemple comme indiqué sur le diagramme de la figure 6.

Quatre états sont possibles : $\delta_j = 0$ détecte le silence, ou absence de parole ; $\delta_j=2$ détecte la présence d'une activité vocale ; et les états $\delta_j = 1$ et $\delta_j = 3$ sont des états intermédiaires de montée et de descente. Lorsque l'automate est dans l'état de silence ($\delta_{n-1,j} = 0$), il y reste si $d_{n,j}$ dépasse un premier seuil $\alpha 1_j$, et il passe dans l'état de montée dans le cas contraire. Dans l'état de montée ($\delta_{n-1,j} = 1$), il revient dans l'état de silence si $d_{n,j}$ dépasse un second seuil $\alpha 2_j$; et il passe dans l'état de parole dans le cas contraire. Lorsque l'automate est dans l'état de parole ($\delta_{n-1,j} = 2$), il y reste si $snr_{n,j}$ dépasse un troisième seuil $\alpha 3_j$, et il passe dans l'état de descente dans le cas contraire. Dans l'état de descente ($\delta_{n-1,j} = 3$), l'automate revient dans l'état de parole si $snr_{n,j}$ dépasse un quatrième seuil $\alpha 4_j$, et il revient dans l'état de silence dans le cas contraire. Les seuils $\alpha 1_j$, $\alpha 2_j$, $\alpha 3_j$ et $\alpha 4_j$ peuvent être optimisés séparément pour chacune des bandes de fréquences j.

Il est également possible que le module 21 fasse interagir les automates relatifs aux différentes bandes.

En particulier, il peut forcer à l'état de parole les automates relatifs à

REVENDICATIONS

1. Procédé de détection d'activité vocale dans un signal de parole numérique (s) dans au moins une bande de fréquences, caractérisé en ce qu'on détecte l'activité vocale sur la base d'une analyse comprenant une comparaison, dans ladite bande de fréquences, de deux versions différentes du signal de parole dont l'une au moins est une version débruitée.

2. Procédé selon la revendication 1, dans lequel ladite comparaison porte sur des énergies respectives ($E_{1,n,j}$, $E_{2,n,j}$), évaluées dans ladite bande de fréquences, des deux versions différentes du signal de parole, ou sur une fonction monotone desdites énergies.

3. Procédé selon la revendication 1 ou 2, dans lequel ladite analyse comprend en outre un lissage temporel de l'énergie ($E_{1,n,j}$) d'une desdites versions du signal de parole, et une comparaison entre l'énergie de ladite version et l'énergie lissée ($\overline{E}_{1,n,j}$).

4. Procédé selon la revendication 3, dans lequel la comparaison entre l'énergie de ladite version ($E_{1,n,j}$) et l'énergie lissée ($\overline{E}_{1,n,j}$) contrôle les transitions d'un automate de détection d'activité vocale d'un état de parole vers un état de silence, tandis que la comparaison des deux versions différentes du signal de parole contrôle les transitions de l'automate de détection de l'état de silence vers l'état de parole.

5. Procédé selon l'une quelconque des revendications 1 à 4, dans lequel les deux versions différentes du signal de parole sont deux versions débruitées par soustraction spectrale non-linéaire, une première des deux versions ($\hat{E}p_{1,n,i}$) étant débruitée de façon à ne pas être inférieure, dans le domaine spectral, à une première fraction ($\beta 1_i$) d'une estimation à long terme ($\hat{B}_{n,i}$) représentative d'une composante de bruit comprise dans le signal de parole, et la seconde des deux versions ($\hat{E}p_{2,n,i}$) étant débruitée de façon à ne pas être inférieure, dans le domaine spectral, à une seconde fraction ($\beta 2_i$) de ladite estimation à long terme, plus petite que la première fraction.

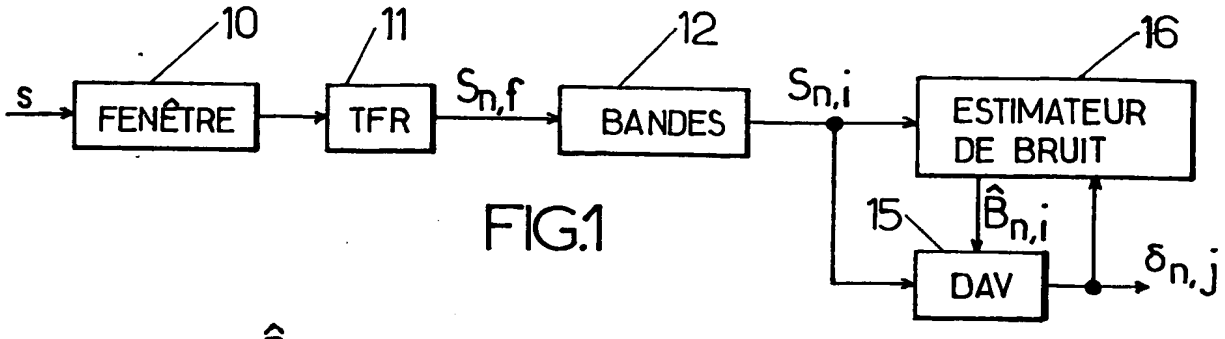


FIG.1

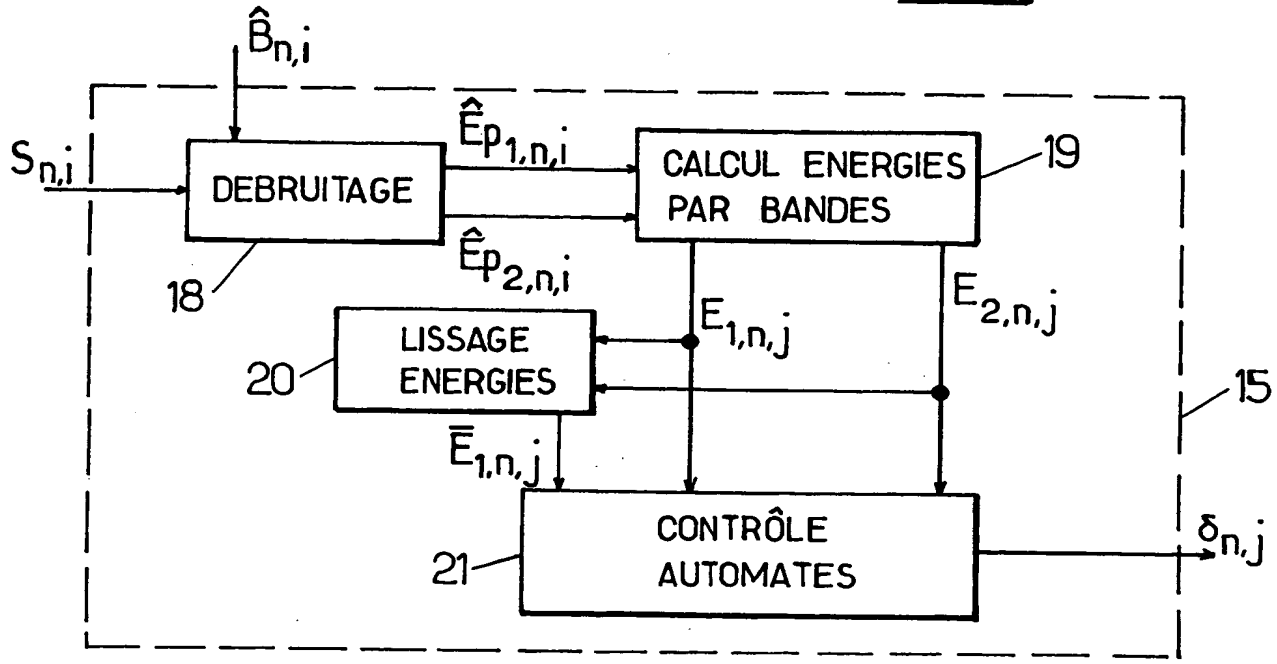
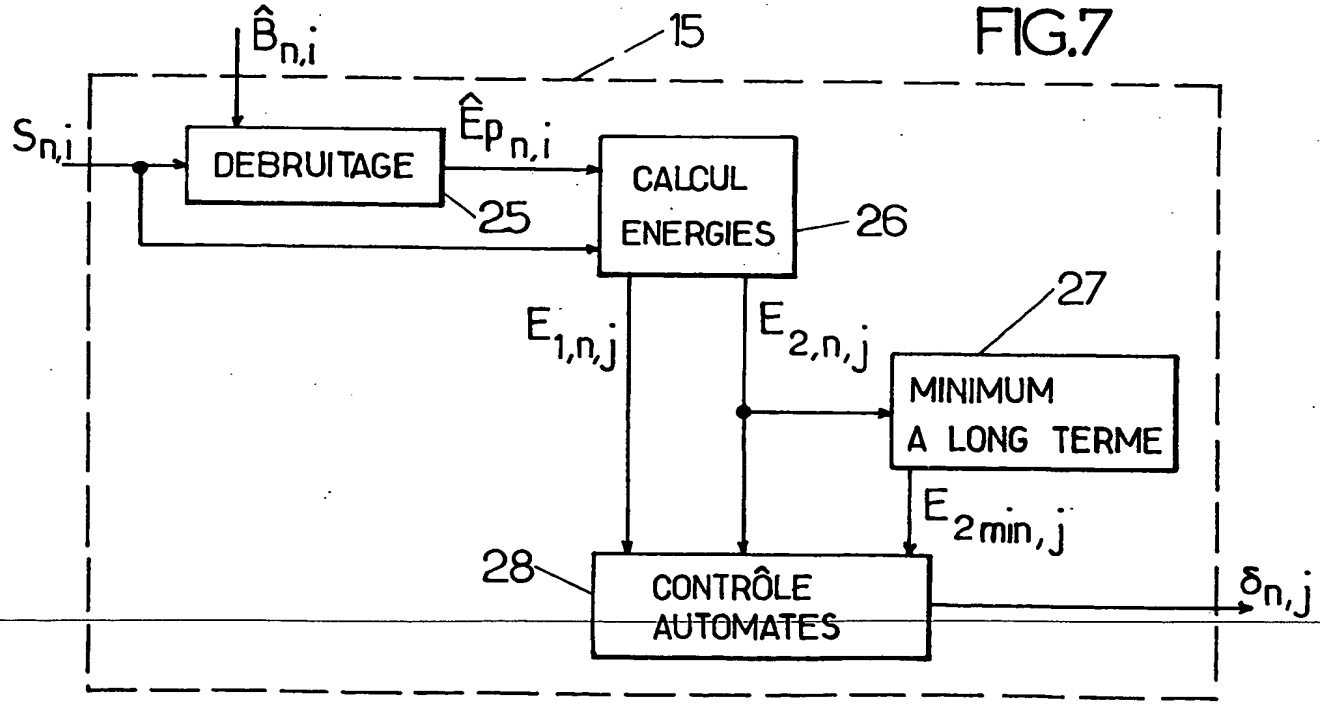


FIG.2

FIG.7



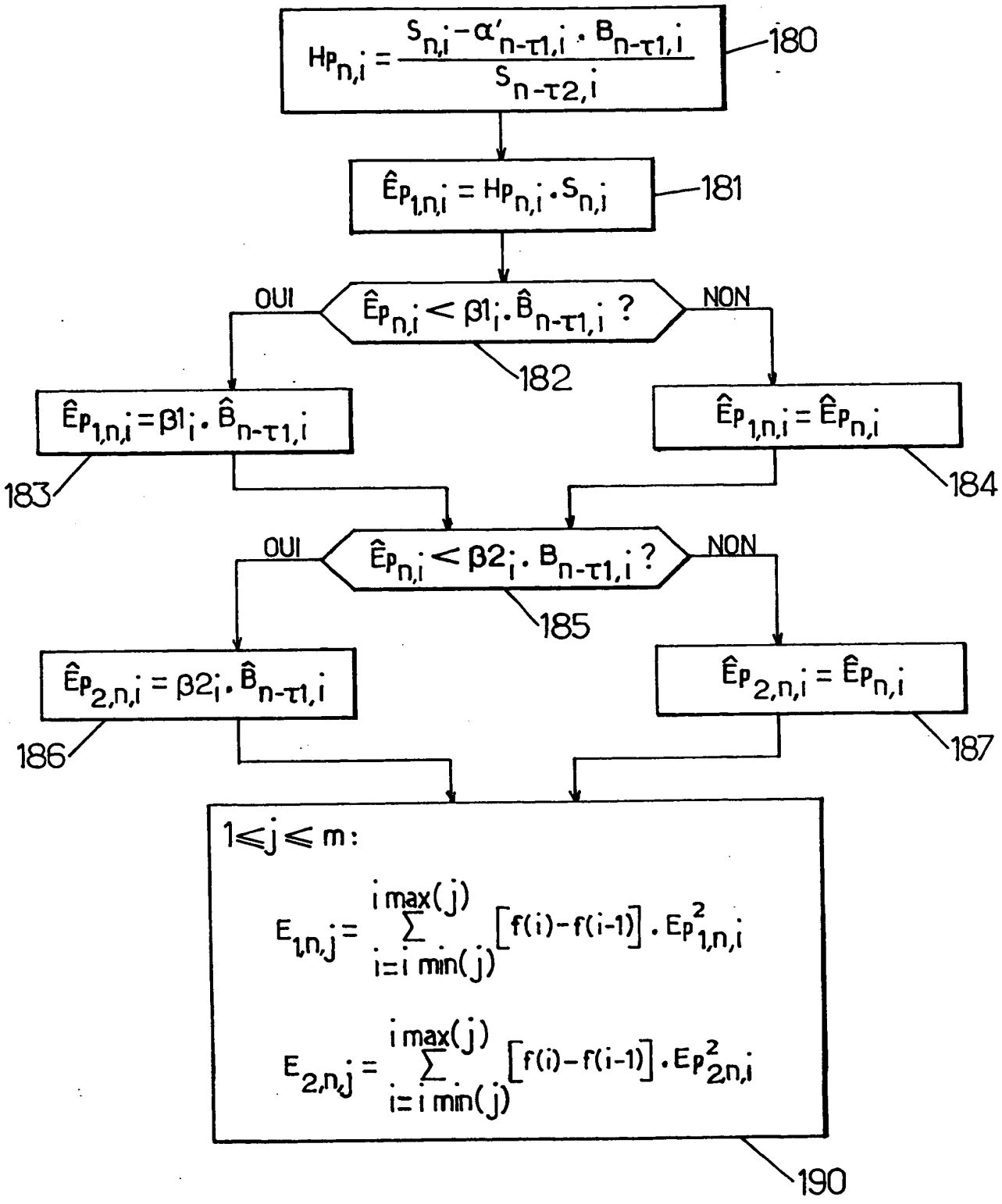

$Hp_{n,i} = \frac{S_{n,i} - \alpha'_{n-\tau1,i} \cdot B_{n-\tau1,i}}{S_{n-\tau2,i}}$
180
$\hat{E}p_{1,n,i} = Hp_{n,i} \cdot S_{n,i}$
181
OUI
$\hat{E}p_{n,i} < \beta1_i \cdot \hat{B}_{n-\tau1,i}$?
NON
182
$\hat{E}p_{1,n,i} = \beta1_i \cdot \hat{B}_{n-\tau1,i}$
183
$\hat{E}p_{1,n,i} = \hat{E}p_{n,i}$
184
OUI
$\hat{E}p_{n,i} < \beta2_i \cdot B_{n-\tau1,i}$?
NON
185
$\hat{E}p_{2,n,i} = \beta2_i \cdot \hat{B}_{n-\tau1,i}$
186
$\hat{E}p_{2,n,i} = \hat{E}p_{n,i}$
187
$1 \leq j \leq m$:
$E_{1,n,j} = \sum_{i=i\,min(j)}^{i\,max(j)} [f(i) - f(i-1)] \cdot Ep^2_{1,n,i}$
$E_{2,n,j} = \sum_{i=i\,min(j)}^{i\,max(j)} [f(i) - f(i-1)] \cdot Ep^2_{2,n,i}$
190

FIG.3

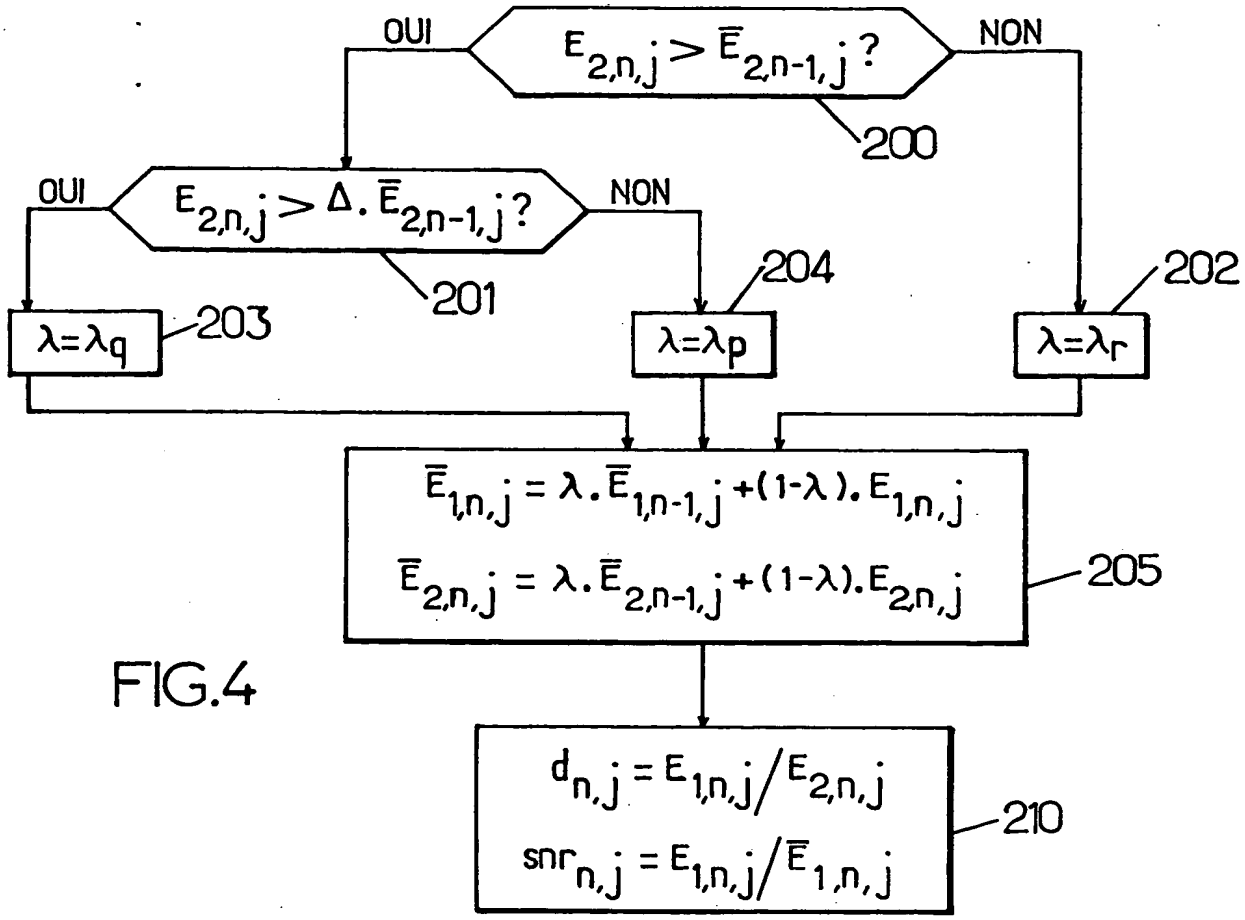
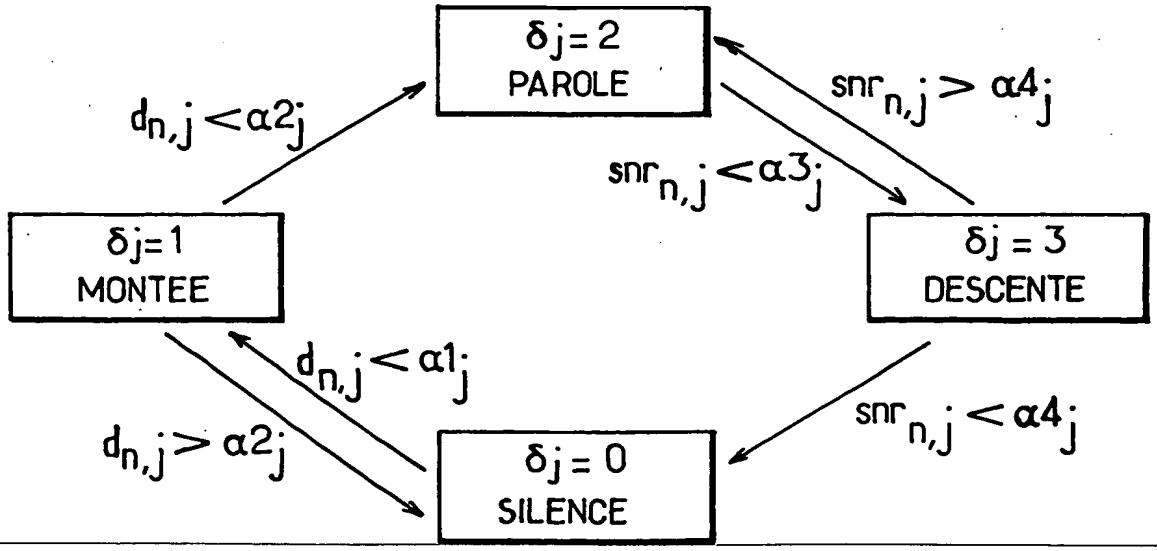

OUI
$E_{2,n,j} > \bar{E}_{2,n-1,j}$?
NON
200
OUI
$E_{2,n,j} > \Delta . \bar{E}_{2,n-1,j}$?
NON
201
204
202
$\lambda=\lambda_q$
203
$\lambda=\lambda_p$
$\lambda=\lambda_r$
$\bar{E}_{1,n,j} = \lambda . \bar{E}_{1,n-1,j} + (1-\lambda) . E_{1,n,j}$
$\bar{E}_{2,n,j} = \lambda . \bar{E}_{2,n-1,j} + (1-\lambda) . E_{2,n,j}$
205
$d_{n,j} = E_{1,n,j} / E_{2,n,j}$
$snr_{n,j} = E_{1,n,j} / \bar{E}_{1,n,j}$
210

FIG.4

FIG.6


$\delta_j = 2$
PAROLE
$d_{n,j} < \alpha 2_j$
$snr_{n,j} > \alpha 4_j$
$snr_{n,j} < \alpha 3_j$
$\delta_j = 1$
MONTEE
$\delta_j = 3$
DESCENTE
$d_{n,j} < \alpha 1_j$
$d_{n,j} > \alpha 2_j$
$snr_{n,j} < \alpha 4_j$
$\delta_j = 0$
SILENCE

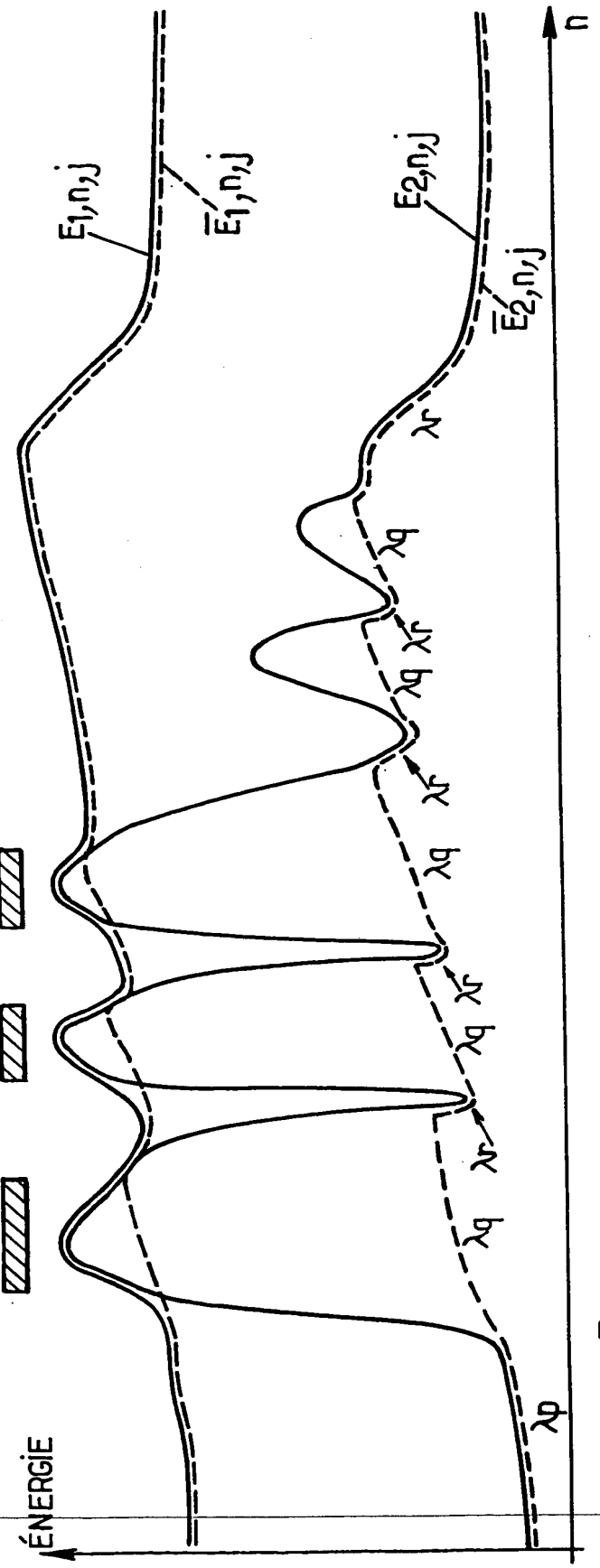


FIG.5.

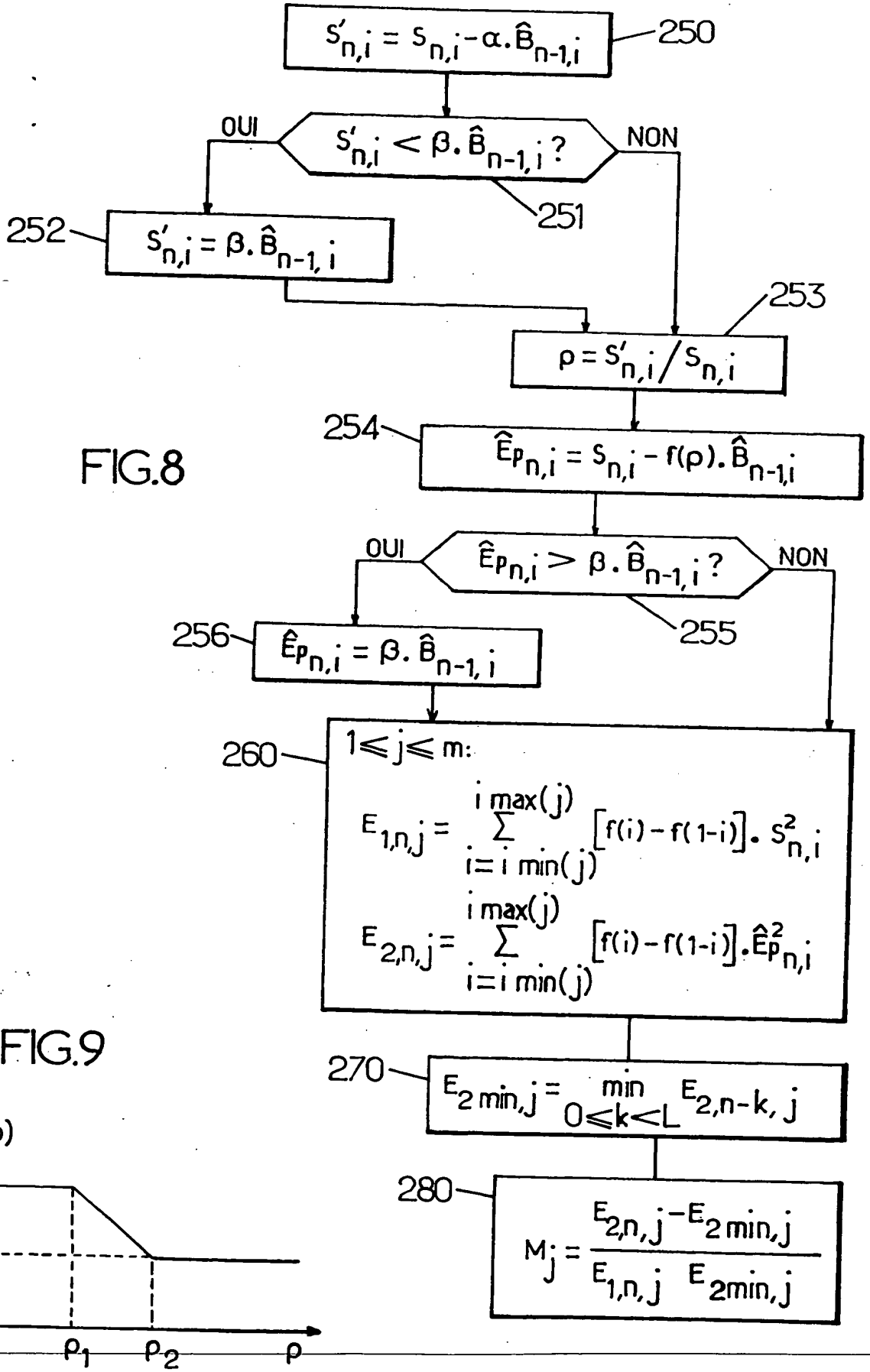

$S'_{n,i} = S_{n,i} - \alpha . \hat{B}_{n-1,i}$
250
OUI
$S'_{n,i} < \beta . \hat{B}_{n-1,i}$?
NON
251
252
$S'_{n,i} = \beta . \hat{B}_{n-1,i}$
253
$\rho = S'_{n,i} / S_{n,i}$
254
$\hat{E}p_{n,i} = S_{n,i} - f(\rho) . \hat{B}_{n-1,i}$
OUI
$\hat{E}p_{n,i} > \beta . \hat{B}_{n-1,i}$?
NON
255
256
$\hat{E}p_{n,i} = \beta . \hat{B}_{n-1,i}$
260
$1 \leqslant j \leqslant m$:
$E_{1,n,j} = \sum_{i=i\min(j)}^{i\max(j)} [f(i) - f(1-i)] . S^2_{n,i}$
$E_{2,n,j} = \sum_{i=i\min(j)}^{i\max(j)} [f(i) - f(1-i)] . \hat{E}p^2_{n,i}$
270
$E_{2\min,j} = \min_{0 \leqslant k < L} E_{2,n-k,j}$
280
$M_j = \frac{E_{2,n,j} - E_{2\min,j}}{E_{1,n,j} \; E_{2\min,j}}$

FIG.8

FIG.9

f(ρ)
2
1
0
ρ1
ρ2
ρ

REVENDICATIONS

1. Procédé de détection d'activité vocale dans un signal de parole numérique (s) dans au moins une bande de fréquences, caractérisé en ce qu'on détecte l'activité vocale sur la base d'une analyse comprenant une comparaison, dans ladite bande de fréquences, de deux versions différentes du signal de parole dont l'une au moins est une version débruitée obtenue en tenant compte d'estimations du bruit compris dans le signal.

2. Procédé selon la revendication 1, dans lequel ladite comparaison porte sur des énergies respectives ($E_{1,n,j}$, $E_{2,n,j}$), évaluées dans ladite bande de fréquences, des deux versions différentes du signal de parole, ou sur une fonction monotone desdites énergies.

3. Procédé selon la revendication 1 ou 2, dans lequel ladite analyse comprend en outre un lissage temporel de l'énergie ($E_{1,n,j}$) d'une desdites versions du signal de parole, et une comparaison entre l'énergie de ladite version et l'énergie lissée ($\overline{E}_{1,n,j}$).

4. Procédé selon la revendication 3, dans lequel la comparaison entre l'énergie de ladite version ($E_{1,n,j}$) et l'énergie lissée ($\overline{E}_{1,n,j}$) contrôle les transitions d'un automate de détection d'activité vocale d'un état de parole vers un état de silence, tandis que la comparaison des deux versions différentes du signal de parole contrôle les transitions de l'automate de détection de l'état de silence vers l'état de parole.

5. Procédé selon l'une quelconque des revendications 1 à 4, dans lequel les deux versions différentes du signal de parole sont deux versions débruitées par soustraction spectrale non-linéaire, une première des deux versions ($\hat{E}p_{1,n,i}$) étant débruitée de façon à ne pas être inférieure, dans le domaine spectral, à une première fraction ($\beta 1_i$) d'une estimation à long terme ($\hat{B}_{n,i}$) représentative d'une composante de bruit comprise dans le signal de parole, et la seconde des deux versions ($\hat{E}p_{2,n,i}$) étant débruitée de façon à ne pas être inférieure, dans le domaine spectral, à une seconde fraction ($\beta 2_i$) de ladite estimation à long terme, plus petite que la première fraction.